

## **REMARKS**

### **I. Drawings**

The Examiner requested corrected drawings because “submitted drawing FIG. 1 is informal.” The Applicant has included a Replacement Sheet for FIG. 1 with this Response.

### **II. Claim Amendments**

The Applicant has amended the claims as follows.

Independent claim 1 has been amended to clarify that the process for modifying an application computer program comprises “incorporating into said application computer program an execution controller that is configured to execute as a debugger in a second electronic execution environment distinct from said first electronic execution environment.” Support for this amendment may be found in the Specification, at least at paragraphs [0006], [0021], [0022], and [0035]; FIG. 2 at reference 40; FIG. 3 at reference 56; and FIG. 4 at FIG. 4 at references 1005 and 1010.

Independent claim 1 has also been amended to clarify that the process for modifying an application computer program comprises “modifying said subsection of said application computer program to a form which, when executed within the first electronic execution environment, causes a transfer or execution control to said execution controller.” Support for this amendment may be found in the Specification, at least at paragraphs [0006], [0035], [0037], [0039], and [0046]; FIG. 2 at reference 37; and FIG. 4 at reference 1025.

Claim 3 has been cancelled.

Dependent claim 4 has been amended to clarify that the step of modifying said subsection includes “a step of adding an instruction that causes a transfer of execution control to said execution controller.” Support for this amendment may be found in the Specification, at least at

paragraphs [0006], [0030] and [0039]; and FIG. 2 at reference 37.

Dependent claim 5 has been amended to be consistent with claim 1.

Dependent claim 7 has been amended to clarify that the step of modifying said subsection includes “a step of adding functionality for the execution controller to communicate with a remote process, wherein the remote process does not execute within either the first or second execution environments.” Support for this amendment may be found in the Specification, at least at paragraphs [0040] and [0041]; and FIG. 1 at references 18 and 20.

Dependent claim 10 has been amended to clarify that the “execution controller communicates information about execution of said application computer program.” Support for this amendment may be found in the Specification, at least at paragraph [0044]; and FIG. 1 at reference 20.

Dependent claim 11 has been amended to be consistent with claim 1.

Independent claim 12 has been amended to clarify that the application computer program has “an executable portion in a form that can be executed in first execution environment under the control of the operating system and a non-executable portion in a non-executable form.” Support for this amendment may be found in the Specification, at least at paragraphs [0004], [0027], and [0028]; and FIG. 2 at references 32.

Independent claim 12 has also been amended to clarify that the execution controller is ‘in a form that can be executed as a debugger in a second execution environment distinct from said first execution environment.’ Support for this amendment may be found in the Specification, at least at paragraphs [0006], [0035], and [0036]; and FIG. 3 at reference 56.

Independent claim 12 has also been amended to clarify that the execution controller “converts the non-executable portions of the application computer program into a form that can

be executed in the first execution environment.” Support for this amendment may be found in the Specification, at least at paragraphs [0007] and [0045]; and FIG. 4 at references 1055.

Claim 13 has been amended to change its dependency.

Dependent claim 14 has been amended to clarify that the “execution controller communicates with a first remote process, and wherein the first remote process does not execute within the first execution environment or the second execution environment.” Support for this amendment may be found in the Specification, at least at paragraphs [0040] and [0041]; and FIG. 1 at reference 18.

Dependent claims 15 and 16 have been amended to be consistent with claim 14.

Dependent claim 17 has been amended to clarify that the execution controller communicates information about the execution of said application computer program to a second remote process, wherein the second remote process does not execute within the first execution environment or the second execution environment.” Support for this amendment may be found in the Specification, at least at paragraph [0044]; and FIG. 1 at reference 20.

Dependent claim 18 has been amended to be consistent with claim 12.

Independent claim 19 has been amended to clarify that the application computer program has “an executable portion in a form that can be executed in a first execution environment under the control of the operating system and a non-executable portion in a non-executable form.” Support for this amendment may be found at least at paragraphs [0004] and [0028]; and FIG. 2 at references 32.

Independent claim 19 has also been amended to clarify that the execution controller is “in a form that can be executed as a debugger in a second execution environment distinct from the first execution environment.” Support for this amendment may be found at least at paragraphs

[0006], [0021], [0022], and [0035]; FIG. 2 at reference 40; FIG. 3 at reference 56; and FIG. 4 at references 1005 and 1010.

Dependent claim 20 has been cancelled.

Dependent claim 21 has been amended to be consistent with claim 19.

Dependent claim 22 has been amended to clarify that “the execution controller communicates with a first remote process, wherein the first remote process does not execute under the control of either the application computer program or the execution controller.”

Support for this amendment may be found at least at paragraphs [0040] and [0041]; and FIG. 1 at reference 18.

Dependent claim 23 has been amended to be consistent with claim 22.

Dependent claim 24 has been amended to clarify that “the execution controller communicates information about execution of the application control program to a second remote process, wherein the second remote process does not execute within the first execution environment or the second execution environment.” Support for this amendment may be found at least at paragraph [0044]; and FIG. 1 at reference 20.

No new matter has been added.

### **III. Claim Objections**

The Examiner objected to claim 13 because of an incorrect dependency. Claim 13 has been amended as discussed above.

The Examiner objected to claims 7, 14 and 22 as confusing. The Applicant has amended claims 7, 14 and 22 as described above.

### **IV. Claim Rejections under 35 U.S.C. 101**

The Examiner rejected claims 12-18 and 19-25 under 35 U.S.C. 101 “because the

claimed invention is directed to non-statutory subject matter. The terms 'being operable to convert' in claim 12 and 'capable of' in claims 14 and 19 fail to provide a physical transformation, or useful, concrete and tangible result."

The Applicant has amended claims 12, 14 and 19, as described above.

## **V. Claim Rejections under 35 U.S.C. 102**

The Examiner has rejected claims 1-25 under 35 U.S.C. 102(e) as being anticipated by US Patent Application Publication 2005/0204348 A1 to Horning et al.

The Applicant respectfully disagrees and traverses the rejections.

"A patent is invalid for anticipation when the same device or method, having all of the elements contained in the claimed limitations, is described in a single prior art reference."

*Crown Operations International, Ltd. v. Solutia, Inc.*, 289 F.3d 1367, 1375 (Fed. Cir. 2002).

### **A. Independent claim 1 is patentable over Horning**

The Applicant respectfully contends that independent claim 1 is patentable over the cited reference because Horning does not disclose at least two elements of claim 1.

1. Horning does not disclose "incorporating into said application computer program *an execution controller that is configured to execute as a debugger* within a second electronic execution environment different from said first electronic execution environment" and "modifying said subsection of said application computer program to a form which, when executed within the first electronic execution environment, *causes a transfer of execution control to said execution controller*."

As supported by the Specification, claim 1 of the preferred embodiment of the present invention includes the step of "incorporating into said application computer program *an execution controller that is configured to execute as a debugger* within a second electronic execution environment different from said first electronic execution environment" and "modifying said subsection of said application computer program to a form which, when executed within the first electronic execution environment, *causes a transfer of execution control to said execution controller*."

*to said execution controller.”* See Specification, at paragraph [0006], “The Execution Controller interacts with the operating system *as a debugger* and controls the execution of the application.” See also Specification, at paragraph [0035], “The secondary thread 54 in turn spawns a new process 56 for the Execution Controller. The Execution Controller immediately attaches to the Main Process 50 *as a debugger;*” FIG. 2 at reference 40; FIG. 3 at reference 56; and FIG. 4 at references 1005 and 1010.

Horning does not disclose “*an execution controller that is configured to execute as a debugger*” or “*caus[ing] a transfer of execution control to said execution controller*.” Rather, in direct contrast to a preferred embodiment of the present invention, a specific function of Horning is to *prevent* an application computer program from executing under a debugger. See Horning, at paragraph [0226], “To keep an attacker from obtaining useful information about an SDP [self-defensive program], *it is advantageous to prevent the SDP’s processes from being run under a debugger.*”

Horning discloses a set of “Obfuscation Tools,” defined as “a variety of tools and techniques [that] can be used to perform obfuscating transformations on a target computer program.” Horning, at paragraphs [0086] and [0087]. “Illustrative types of attacks...include...3. *Running code under a debugger*, setting breakpoints, single-stepping, and examining memory and registers.”

Horning further discloses “techniques” for defending *against* the use of a debugger. See Horning, at paragraph [0021], “FIG. 9 illustrates a technique for interfering with a debugger”; paragraph [0022], “FIG. 10 illustrates a technique for detecting the presence of a debugger....”; and paragraph [0023], “FIG. 11 illustrates a technique for interfering with the operation of a debugger by storing instructions and/or data into the instruction stream.” See also Horning, at

paragraph [0118], “Because certain debuggers are detectable, either directly or by obtaining information from the operating system, *the program can also be configured to determine if a debugger is running, and if so, to take defensive action.*” See also Horning, at paragraph [0123] and [0127], “Local Obfuscation Techniques...[include]...3. Techniques that discourage dynamic analysis. These techniques focus on making dynamic analysis of the code more difficult by *confusing what debuggers and/or related tools see at runtime.*”

The Examiner has referenced Horning at paragraph [0594] for support. See March 23, 2007, Office Action, at page 5, “Per claim 3:...[0594], provide internal tracing and debugging features.” The Applicant respectfully disagrees.

Horning at paragraph [0594] is part of the section that discloses “Debugging SDPs.” See Horning, at paragraph [0591]. This section of Horning discloses techniques that can be used *instead of a debugger*, “Once software self-defense techniques have been applied, the resulting program may be more difficult to debug. If debugging an SDP is required, there are several measures that can be applied to make it more practical.”

The Applicant respectfully contends that “debugging” a computer program by the various techniques suggested in Horning is separate and distinct from “incorporating into said application computer program *an execution controller that is configured to execute as a debugger* within a second electronic execution environment different from said first electronic execution environment” and “modifying said subsection of said application computer program to a form which, when executed within the first electronic execution environment, *causes a transfer of execution control to said execution controller.*”

In summary, then, Horning does not disclose at least two elements of independent claim 1, and claim 1 is patentable over the reference.

**B. Dependent claims 2 and 4-11 are patentable over Horning.**

Claims 2 and 4-11, which depend from claim 1, are patentable for at least the same reasons discussed above.

**C. Independent claim 12 is patentable over Horning.**

The Applicant respectfully contends that independent claim 12 is patentable over the cited reference because Horning does not disclose at least one element of claim 12.

1. Horning does not disclose “*an execution controller in a form that can be executed as a debugger in a second execution environment.*”

As supported by the Specification, and as discussed with reference to claim 1, above, the preferred embodiment of the present invention includes “*an execution controller in a form that can be executed as a debugger in a second execution environment.*” See Specification, at paragraph [0006], “The Execution Controller interacts with the operating system *as a debugger* and controls the execution of the application.” See also Specification, at paragraph [0035], “The secondary thread 54 in turn spawns a new process 56 for the Execution Controller. The Execution Controller immediately attaches to the Main Process 50 *as a debugger*,” FIG. 2 at reference 40; FIG. 3 at reference 56; and FIG. 4 at references 1005 and 1010.

Horning does not disclose “*an execution controller in a form that can be executed as a debugger.*” Rather, as discussed above with reference to claim 1, and in direct contrast to a preferred embodiment of the present invention, a specific function of Horning is to *prevent* an application computer program from executing under a debugger. See Horning, at paragraph [0226], “To keep an attacker from obtaining useful information about an SDP [self-defensive program], *it is advantageous to prevent the SDP’s processes from being run under a debugger.*”

In summary, then, Horning does not disclose at least one element of independent claim 12, and claim 12 is patentable over the reference.

**D. Independent claim 19 is patentable over Horning.**

The Applicant respectfully contends that independent claim 19 is patentable over the cited reference because Horning does not disclose at least one element of claim 19.

**1. Horning does not disclose “launching an execution controller, said *execution controller* in a form that can be executed as a debugger in a second execution environment.”**

As supported by the Specification, and as discussed with reference to claim 1, above, the preferred embodiment of the present invention includes “*an execution controller in a form that can be executed as a debugger* in a second execution environment.” See Specification, at paragraph [0006], “The Execution Controller interacts with the operating system *as a debugger* and controls the execution of the application.” See also Specification, at paragraph [0035], “The secondary thread 54 in turn spawns a new process 56 for the Execution Controller. The Execution Controller immediately attaches to the Main Process 50 *as a debugger*;” and FIG. 2 at reference 40; FIG. 3 at reference 56; and FIG. 4 at references 1005 and 1010.

Horning does not disclose “*an execution controller in a form that can be executed as a debugger*.” Rather, as discussed above with reference to claim 1, and in direct contrast to a preferred embodiment of the present invention, a specific function of Horning is to *prevent* an application computer program from executing under a debugger. See Horning, at paragraph [0226], “To keep an attacker from obtaining useful information about an SDP [self-defensive program], *it is advantageous to prevent the SDP’s processes from being run under a debugger.*”

In summary, then, Horning does not disclose at least one element of independent claim 19, and claim 19 is patentable over the reference.

**V. Conclusion**

The Applicant has addressed each of the Examiner’s objections and rejections, and respectfully contends that claims 1-2, and 4-25 are patentable over the cited prior art. The

Applicant respectfully requests reconsideration and allowance.

If for any reason this Response is found to be incomplete, or if at any time it appears that a telephone conference with counsel would help advance prosecution, please telephone the undersigned in Westborough, Massachusetts, (508) 898-1501. If any payment during prosecution is found to be insufficient or if any overpayment is found, please charge any deficiency or credit any overpayment to my deposit account number 50-1582.

Respectfully submitted,



---

Brian M. Dingman  
Attorney for Applicant  
Reg. No. 32,729  
Customer No. 25834  
Tel: (508) 898-1501  
Fax: (508) 898-1502

MIRICK, O'CONNELL, DEMALLIE & LOUGEE, LLP  
1700 West Park Drive, Westborough, MA 01581